Enabling IoT Ecosystems through Platform Interoperability

Arne Broering[1]
Stefan Schmid[2]
Corina-Kim Schindhelm[1]
Abdelmajid Khelil[3]
Sebastian Kaebisch[1]
Denis Kramer[3]
Danh Le Phouc[4]
Jelena Mitic[1]
Darko Anicic[1]
Ernest Teniente[5]

[1] Siemens AG – Munich, Germany
[2] Bosch Corporate Research – Stuttgart, Germany
[3] Bosch Software Innovations – Stuttgart, Germany
[4] NUI Galway, Insight Centre – Galway, Ireland
[5] Universitat Politècnica de Catalunya – Barcelona, Spain

Abstract

Today, the Internet of Things (IoT) is comprised of vertically oriented platforms for things. Developers who want to use them need to negotiate access individually and need to adapt to the platform-specific API and information models. Having to do these efforts for each platform often outweighs the possible gains for application developers to adapt their applications to multiple platforms. This fragmentation of the IoT and the missing interoperability result in high entry barriers for developers and currently prevent the emergence of broadly accepted IoT ecosystems. This article presents the work of the BIG IoT project that aims at igniting an IoT ecosystem as part of the European Platform Initiative (IoT EPI). We introduce an architectural model for IoT ecosystems, and highlight five common interoperability patterns that need to be supported for enabling cross-platform interoperability and establishing successful IoT ecosystems.

Keywords

D.2.12 Interoperability, C.2.0.a Architecture, D.2.11.e Patterns, D.3.1.a Semantics

The Problem of Missing Interoperability on the IoT

The idea of an Internet of Things (IoT) is no more a futuristic vision, but indeed an increasing reality that reaches to various application domains ranging from quantified self and smart home applications, over smarter cities and eHealth systems, to Industry 4.0. Dozens of IoT platforms are upcoming. These include cloud solutions, such as Evrythng[1], ThingWorx[2], Xively[3], or Yaler[4], but also on premise solutions such as Bosch's IoT Suite[5], as well as thematically or geographically focused platforms, such as the Smart Data platform for the Piedmont region[6]. However, up to now, these IoT platforms failed to form vibrant IoT ecosystems. This is mainly due to the large number of stakeholders involved in IoT ecosystems, such as *platform providers*, *thing providers*, *developers*, and *users*.

Platform providers include startups, large companies, or public institutions, e.g., traffic management agencies and public transportation providers. *Thing providers* are enterprises or administrations that operate thingsthings. Usually, they require an IoT platform to manage their thingsthings. In some cases, they rely on an external platform provider. Typically, a thing provider acts as *user* of the IoT platform. *Developers* are individuals or companies that develop services or applications based on the platforms and the things they manage. Services and applications can provide functionality for enterprises or administrations, or for mobile devices that are utilized by the *users*.

Currently, there is no vibrant collaborative IoT ecosystem, since the entry barriers are high and the potential gain is low for a single stakeholder. Providers of platforms, things, and services require a simple, established way to sell the access to their assets. Marketplaces that enable providers to monetize access to their things, platforms and services are not yet available. Once these marketplaces are established, developers will be able to easily build IoT services and applications and build their products around these. Revenue streams can then be shared across all contributing entities (i.e., service providers, platform providers, and thing providers). A key task of a marketplace is to provide extended functionalities to enable the advertising, dynamic discovery, automated orchestration, and negotiation of services to facilitate their usage.

While marketplaces will play a key role in the monetization of services and IoT assets, still a serious entry barrier to IoT ecosystems needs to be tackled before marketplaces can bring their effect: the *lack of interoperability* across IoT platforms and things. Today, we are dealing with various vertically-oriented and mostly closed systems. Architectures for IoT are built on heterogeneous standards (e.g., IETF CoAP [1], OASIS MQTT [2], OMA LWM2M [3], OGC SWE [4], or OneM2M [5]) or even proprietary interfaces. As a result, most existing and emerging IoT platforms offer heterogeneous ways for accessing things and their data. This causes interoperability problems when overarching, cross-platform, and cross-domain applications are to be built, and eventually prevents the emergence of vibrant IoT ecosystems. Additionally, it leads to barriers for business opportunities, especially for small innovative enterprises, which cannot afford to provide their solution across multiple platforms. They can only provide applications and services for a small number of platforms, e.g., a traffic information application for an IoT platform of a specific city. This lack of interoperability results in lost business opportunities and prevents innovative business ideas.

---

[1] www.evrythng.com
[2] www.thingworx.com
[3] www.xively.com
[4] https://yaler.net/
[5] https://www.bosch-si.com/products/bosch-iot-suite/benefits.html
[6] http://www.smartdatanet.it/presentation.html

Towards an Interoperable IoT Ecosystem

Bridging the Interoperability Gap of the IoT (BIG IoT) is a project[7] that aims at enabling the emergence of cross-platform, cross-standard, and cross-domain IoT services and applications towards building IoT ecosystems. These ecosystems will connect thing and service providers as well as the users of those. This vision is similar to what has been articulated in form of concepts such as pervasive or ubiquitous computing [6], where the physical environment is equipped with computational capabilities. While those concepts are very much focused on implications and new experiences for the user, our notion of IoT ecosystems primarily addresses issues of technical design and realization. Technologically supporting interoperability is in focus, similar to the work presented in [7]. However, while the authors of [7] propose a solution for managing interoperable cloud applications, this work focuses on interoperability in the intermediary of IoT applications, services, and platforms.

An application example within an IoT ecosystem is described in the following: A cross-platform IoT application can access an IoT platform of a user's wearable sensors to automatically deduce that the user is leaving her workplace. Next, the application accesses a smart mobility platform to purchase a ticket for the commuter train and navigates the user to the reserved seat. Further, the application can contact the IoT platform of the user's fridge, to tell her to stop by a supermarket on her way home. Finally, the application accesses a smart home platform to heat up the user's house before arrival. Similarly, a cross-platform IoT application for a smarter workplace could be enabled to access the wearable sensors of users to utilize the gathered data for different purposes, e.g., to monitor the environment at their workplace. The described multi-purpose of things and data gathered by separate IoT platforms can create great benefits.

In order to ignite such an IoT ecosystem, interoperability across platforms needs to be enabled. Once this cross-platform interoperability is achieved, this will allow new applications by combining data from multiple platforms (e.g. parking information from various smart city platforms). Also, platforms from multiple domains can then be combined, e.g., a wearables platform with a smart home platform. An application will work on top of different platforms, e.g., the same application works on top of a smart city platform in Berlin, in Barcelona and in London. Thus, we present an architecture here that aims at overcoming these hurdles through (1) a common Web interface, called the BIG IoT API, (2) semantic descriptions of resources and services, as well as (3) a marketplace as the core driver of the ecosystem, providing functionalities such as authentication, discovery and charging.

Architectural Model of an Interoperable IoT Ecosystem

Figure 1 outlines these key components and how we envision an IoT ecosystem. The different IoT platforms give access to various kinds of things. Additionally to providing their own interfaces, IoT platforms are enriched with a common interface, the BIG IoT API, which offers the required set of functionalities for interoperability with other platforms. IoT platforms can operate either on cloud-level (e.g., server, data center), on fog-level (e.g., gateway, cellular communication base station), or on device-level (e.g., a Raspberry PI, wearable, Smart phone). The core of the BIG IoT API can be mutually used independent of this scale of the platform. Through the common API, it becomes now easier to develop software artifacts as clients of different platforms. Among such software artifacts, we distinguish between *services* and *applications*. While both are consumers of resources (information or functions), services can also act as providers of resources. This enables services to be composed into more complex or

[7] http://big-iot.eu/

added value services. The resources of providers are advertised on the marketplace, for consumers to discover them and to gain access to desired providers. Thereby, we foresee that there will be multiple marketplaces in the future. Marketplaces could be setup per application domain (e.g., for smart city, building automation, or manufacturing) or there could exist multiple marketplaces related to one domain but setup by different organizations. As long as they are compliant to the defined interfaces, those marketplaces can further foster IoT ecosystems.

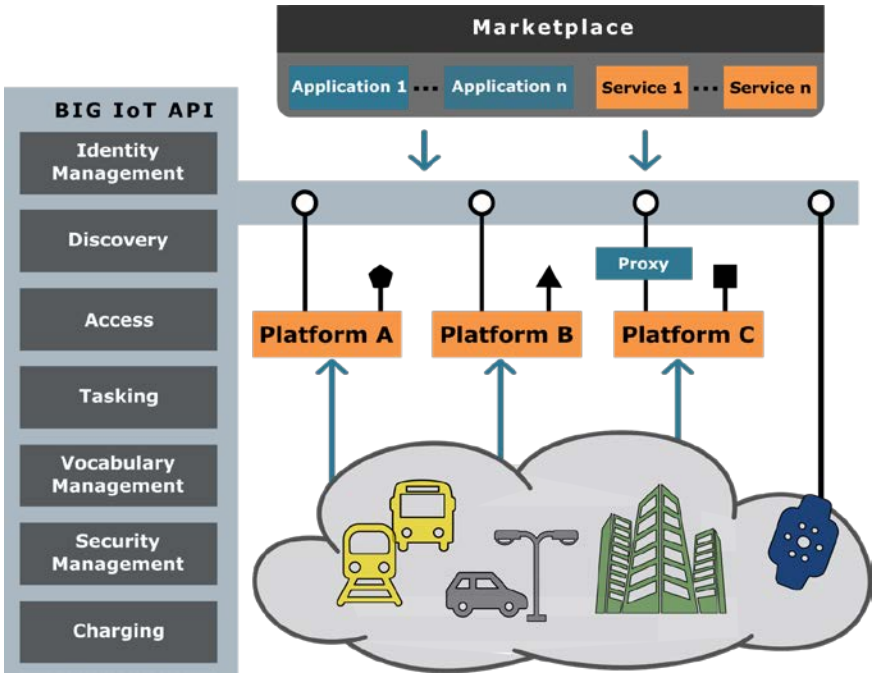

Figure 1: IoT Ecosystem Overview

To enable interoperability for IoT platforms on cloud-, fog-, as well as on device-level, the BIG IoT API offers a well-defined set of functionalities. The key *functionalities* that need to be part of the common API are (a) *Identity management* to enable the registration of resources, (b) *Discovery* of resources according to user defined search criteria, (c) *Access* to meta-data and data (data pull as well as publish/subscribe of data streams), (d) *Tasking* to forward commands to things, (e) *Vocabulary management* for semantic descriptions of concepts, (f) *Security management* including authentication, authorization, and key management, as well as (g) *Charging* that allows the monetization of assets through mechanisms for payment and billing.

Patterns of Interoperability for an IoT Ecosystem

Reaching interoperability on the IoT based on the model described above, requires a closer look at interactions of the different key components. Thereby, interoperability relates to the *syntax* as well as to the *semantics* of interfaces. Syntactic interoperability can be reached through clearly defined and agreed upon data and interface formats as well as encodings. Semantic interoperability can be achieved through commonly agreed information models (e.g., defined with ontologies) of the terms used as part of the interfaces and exchanged data. In Figure 2, we identify five generic interoperability patterns for IoT ecosystems that need to be supported in order to achieve the goal of lowering market entry barriers for developers.

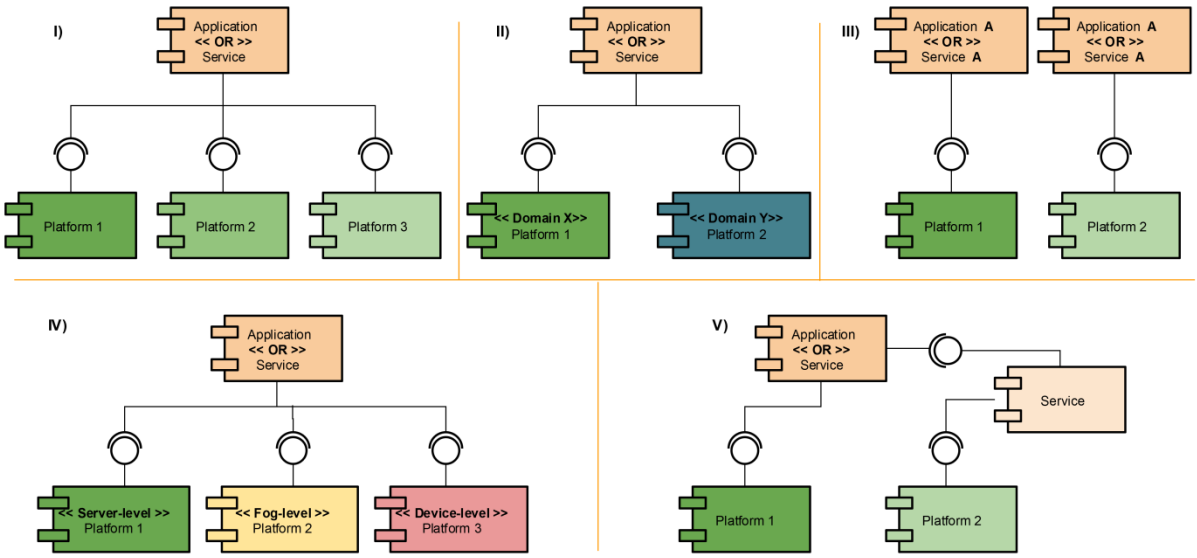

Figure 2: The five patterns of interoperability: I) "Cross Platform Access", II) "Cross Application Domain Access", III) "Platform Independence", IV) "Platform-Scale Independence", and V) "Higher-level Service Facades" Pattern.

The "Cross Platform Access" pattern (Figure 2, I), is the fundamental characteristic of an interoperable IoT ecosystem. The pattern entails that an application or service accesses resources (information or functions) from multiple platforms through the same interface specification. For example, an "air quality monitoring" application gathers information on different air quality indicators such as NO2, CO, and O3 offered by different platforms. The challenge of realizing this pattern lays in allowing applications or services to discover platforms with relevant information, and enabling platforms that are potentially from different providers to expose the same interface and use the same formats to communicate data.

The pattern "Cross Application Domain Access" (Figure 2, II) extends the "Cross Platform Access" pattern. Services/applications access information and functions not only from multiple platforms, but also from platforms which host information from different verticals or application domains. As semantic descriptions of the information sources of a platform can be accessed through the common platform interface, integrating such (originally heterogeneous) data into one service/application becomes possible. An application that gathers data from different domains, could e.g. access air quality information, such as O3, and traffic monitoring information, such as average speed, to provide healthy bicycle routes with cleaner air.

The pattern "Platform Independence" (Figure 2, III) represents another basic characteristic of an interoperable IoT ecosystem. It entails that the identical application or service can be used on top of two different IoT platforms (e.g. in different regions). This can be achieved by allowing an application/service to discover relevant IoT platforms and to interact with the different platforms in a uniform manner. For example, these can be two deployments of a "smart parking" service used for two different geographic regions (e.g., Barcelona and London), which have their own platforms with information about parking spots. Realizing platform independence is particularly challenging, when the information provided by both platforms is created by different kinds of things. For example, in case of parking information, the information on spot availability could be

generated by radar-based sensors mounted on street lamps as well as ultrasound-based sensors in the ground.

The “Platform-Scale Independence” pattern (Figure 2, IV) focuses on integrating platforms of different scale. *Server-level platforms* usually manage a large number of devices (e.g., a cloud platform) and host a vast amount of data. *Fog-level platforms* connect close-by devices (e.g., a gateway) and manage data with limited spatio-temporal scope. *Device-level platforms* grant direct access to things (e.g., a sensor device) and typically host small amounts of data. By implementing this pattern, the platform hides its scale towards connecting services or applications. Data from device-/fog-/server-level platforms can be uniformly used by services/applications. For example, an application displays information on air quality monitoring to the user (e.g., as visualizations on a map). On the one hand, the application could allow accessing aggregated information such as the computed air quality index for a certain region from a server-level IoT platform. On the other hand, the application may additionally enable to access data directly from air quality stations (i.e., fog-level platforms), e.g., to display time series from unadulterated data.

Finally, the pattern “Higher-level Service Facades” (Figure 2, V) extends the interoperability requirements from platforms to higher-level services. The idea is that not only platforms but also services offer information and functions via the common API. Thereby, a service acts as a façade towards an IoT platform and accesses the offered information or functions to provide value-added functionalities. For example, an air quality viewer application can on the one hand access a platform P1 that provides already aggregated air quality data. On the other hand, the application can access a service that aggregates air quality data from platform P2, e.g., because P2 does not have the capabilities to perform data aggregation or host long-term time series data.

Once the above described patterns are implemented, they enable reuse and composition of services as well as easy integration of data from different platforms. Our vision of IoT ecosystems goes even beyond those benefits. Dynamic search and orchestration of information as well as automatic charging are necessary to allow for a flourishing and easy to use ecosystem. Within the ecosystem, as an example, country borders do not matter as long as applications and services are part of the ecosystem. As an example, in case of a smart car parking app, if an end user travels, she would not have to take care of downloading a new parking app for the target country. If data from a platform in the target country is provided based on a compliant semantic framework, the ecosystem could enable an automatic discovery of and connection to the right information sources with the application to allow for seamless usage.

The BIG IoT Architecture

Figure 3 presents an overview of the proposed BIG IoT architecture for IoT ecosystems. An open marketplace for IoT platforms and services as *providers* to trade available resources (information and functions) is at its centre. IoT applications or services as *consumers* of resources can use the marketplace to discover them and access them in real-time. The architecture has been specifically designed to support all of the above-described patterns of interoperability. The architecture is centered around a common set of interfaces, referred to as the BIG IoT API, that are supported both by resource providers and consumers, as well as the marketplace, where resources are traded. These interfaces include the following basic interactions:

M1: Authentication and authorization on the marketplace (resource providers and consumers)
M2: Registration of resources on the marketplace (resource providers)
M3: Discovery of resources on the marketplace (resource consumers)
M4: Accounting of resources provided (resource providers) and resources consumed (consumers)
A1: Access to resources (among resource consumers and providers)

The common API and the marketplace are the basis for resource providers and consumers to discover each other, to communicate and exchange resources and to perform charging and billing. As such, they constitute the basis for enabling interoperability for the patterns I - V.

The main challenge for patterns II, III, and V is that they target interoperability among highly heterogeneous entities, such as a) providers and consumers from different verticals or application domains (II); b) providers hosted on different IoT platforms, e.g. located in different regions (III); and c) providers on different provider systems, e.g. an IoT platform or a service (V). To bridge the interoperability gap for those patterns, the architecture mandates the use of common information models, such as provided by the Semantic Web and Linked Data [8]. Such common information models support providers (platforms or services) to describe the resources they offer in a machine understandable manner, so that consumers (services/applications) of a different domain, region or system can understand and process them. For example, Schema.org vocabularies are the shared common understandings between search engines and billions of web pages [9]. The information models are also used by the marketplace to match providers and consumers based on their supplies and demands. To enable that, data providers and data consumers can share the same vocabularies for “smart object”, “sensor”, “measurement”, etc. in the same way that search engine providers agree with Web developers on how to describe “restaurant”, “hotel”, “airline”, etc.

Important in this figure is also the concepts of the BIG IoT Consumer and Provider Libs. These libraries implement functionalities of the ecosystem. For example, the Provider Lib implements the *Register* interface (M2) to offer resources via the marketplace and offers the *Access* interface (A1) to provide the information to a consumer. The benefit of these libraries is that developers of platforms, services and applications are supported in trading their resources on the marketplace or use the marketplace to discover and access them. They only have to implement once the *Provider* (P1) or *Consumer* (P2) interface and can easily update the libraries in order to further comply in case of changes in the details of the underlying message formats and interactions.

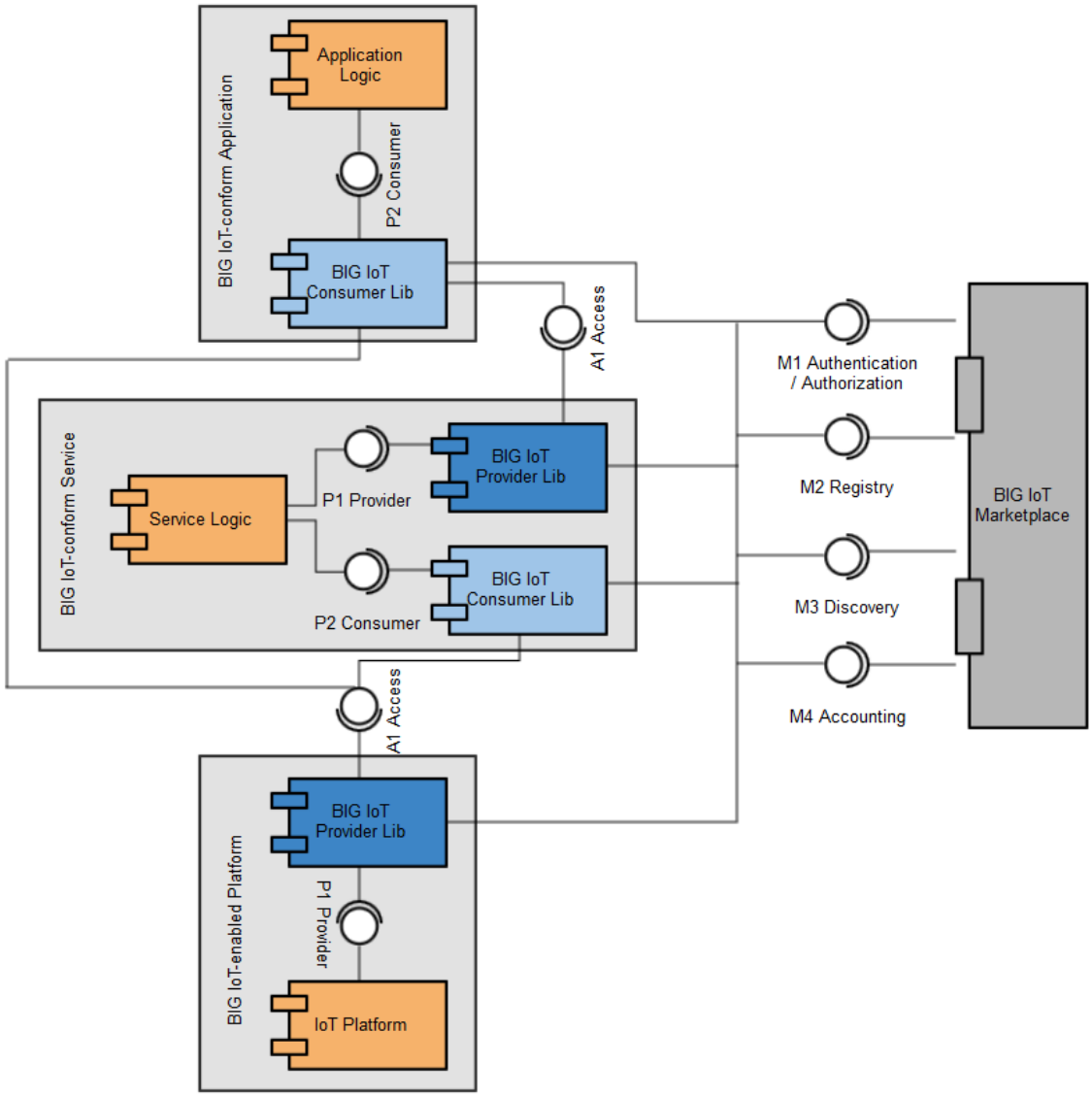

Figure 3: BIG IoT Architecture Overview

Conclusions and Outlook

With more and more devices being connected, the Internet of Things is on the rise. However, today, devices and their data are gathered in vertically oriented IoT platforms. Often these IoT platforms act as closed silos with a very narrow application focus. These platforms promote their specific interface and data formats and typically restrict communication to those formats. This fact is preventing a broadly accepted IoT ecosystem to emerge.

This article presents the model of an IoT ecosystem including five key interoperability patterns, which need to be supported to bridge the interoperability gap of the Internet of Things. We identify three key pillars for such interoperable IoT ecosystems: a common API, well-defined information models, and a marketplace to monetize access to resources. The common and

generic BIG IoT API as well as the used information models are thereby defined in conjunction with the *Web of Things Interest Group* at the W3C[8], to bring the outcomes into community supported standards.

The BIG IoT project (http://big-iot.eu/) has started to ignite an IoT ecosystem with overall 8 IoT platforms that are being equipped with the common API. Among them are productive platforms from Bosch, CSI, Vodafone, and WorldSensing. The use cases to test the interoperability are from the mobility domain including smart parking, bike sharing and traffic management. The use cases are implemented using the IoT platforms and newly developed services and applications. They will be demonstrated and tested in three pilots in Barcelona (Spain), Piedmont (Italy) and Berlin/Wolfsburg (Germany). In order to showcase the realization of the five interoperability patterns, the implemented services and applications will be reused and transferred between pilots.

The ultimate goal of growing this initiated IoT ecosystem, by including more IoT platforms as well as applications and services, requires being attractive for developers. A fundamental risk of our approach is that the developed concepts (e.g., common API and information models) are unattractive for developers as well as platform providers. We are mitigating this risk by including such stakeholders through close interaction (e.g., reaching out through surveys and market studies). Secondly, to activate the ecosystem, BIG IoT follows an approach of openness towards the IoT developer community. This will be underpinned by community outreach (e.g., hackathons), an open development of the API (supported through the W3C Web of Things group), and releasing the developed software as open source. Conceptually and technologically, the project will tackle different issues in the coming months. E.g., a coherent security concept is to be developed for the interaction between marketplace and the various IoT platforms. These generally come with their own authentication and authorization management, which makes a centralized user management for the marketplace difficult. Further, common vocabularies and semantic models have to be developed or reused (as far as there is existing work). Based on these vocabularies, descriptions for IoT platform resources can be defined. In a next step, mechanisms for intelligently composing those resources will be defined to encourage and maximize reuse of existing resources. E.g., this could support an automatic composition of a smart parking service out of a parking finder and a parking reservation service.

Acknowledgments

This work is supported by the BIG IoT project that has received funding from the European Commission's Horizon 2020 research and innovation programme under grant agreement No 688038. This article presents the current status of the authors' work in the project. We thank the consortium partners for their feedback and fruitful discussions. The work will be further evolved as part of the ongoing architecture development in the project.

References

[1] C. Bormann, A. P. Castellani, and Z. Shelby, "CoAP: An Application Protocol for Billions of Tiny Internet Nodes," *IEEE Internet Comput.*, vol. 16, no. 2, pp. 62–67, Mar. 2012.

[2] IBM and Eurotech, "MQTT V3.1 Protocol Specification." [Online]. Available: http://public.dhe.ibm.com/software/dw/webservices/ws-mqtt/mqtt-v3r1.html. [Accessed: 24-

[8] https://www.w3.org/WoT

Apr-2014].
[3] Open Mobile Alliance, “Lightweight Machine to Machine Technical Specification, Candidate,” OMA, 2015.
[4] A. Bröring, J. Echterhoff, S. Jirka, I. Simonis, T. Everding, C. Stasch, S. Liang, and Rob Lemmens, “New Generation Sensor Web Enablement,” *Sensors*, vol. 11, no. 3, pp. 2652–2699, 2011.
[5] J. Swetina, G. Lu, P. Jacobs, F. Ennesser, and J. Song, “Toward a standardized common M2M service layer platform: Introduction to oneM2M,” *Wirel. Commun. IEEE*, vol. 21, no. 3, pp. 20–26, 2014.
[6] M. Weiser, “Some computer science issues in ubiquitous computing,” *Commun. ACM*, vol. 36, no. 7, pp. 75–84, 1993.
[7] N. Loulloudes, C. Sofokleous, D. Trihinas, M. D. Dikaiakos, and G. Pallis, “Enabling interoperable cloud application management through an open source ecosystem,” *IEEE Internet Computing*, vol. 19, no. 3, pp. 54--59, 2015.
[8] C. Bizer, T. Heath, and T. Berners-Lee, “Linked Data - The Story so far,” *Journal on Semantic Web and Information Systems*, vol. 5, no. 3, pp. 1–22, 2009.
[9] R. Guha, D. Brickley, and S. Macbeth, “Schema. org: Evolution of structured data on the web,” *Communications of the ACM*, vol. 59, no. 2, pp. 44–51, 2016.